NAMING THINGS

there are only two hard things in computer science

cache invalidation and

naming things

- Phil Karlton

NOT IDIOMATIC

Unnecessarily verbose

```
func Read(buffer *Buffer, inBuffer []byte) (size int, err error) {
    if buffer.empty() {
        buffer.Reset()
    }

    size = copy(
      inBuffer,
      buffer.buffer[buffer.offset:])

    buffer.offset += size
    return size, nil
}
```

IDIOMATIC

Concise and idiomatic

```
func Read(b *Buffer, p []byte) (n int, err error) {
    if b.empty() {
        b.Reset()
    }

    n = copy(p, b.buf[b.off:])

    b.off += n

    return n, nil
}
```

ABBREVIATIONS

Common abbreviations used in Go

```
var s string                // string
var i int                   // index
var num int                 // number
var msg string              // message
var v string                // value
var val string              // value
var fv string               // flag value
var err error               // error value
var args []string           // arguments
var seen bool               // has seen?
var parsed bool             // parsing ok?
```

ABBREVIATIONS

Common abbreviations used in Go

```
var buf []byte          // buffer
var off int             // offset
var op int              // operation
var opRead int          // read operation
var l int               // length
var n int               // number or number of
var m int               // another number
var c int               // capacity
var c int               // character
var a int               // array
var r rune              // rune
var sep string          // separator
```

ABBREVIATIONS

Common abbreviations used in Go

```
var src int                 // source
var dst int                 // destination
var b byte                  // byte
var b []byte                // buffer
var buf []byte              // buffer
var w io.Writer             // writer
var r io.Reader             // reader
var pos int                 // position

...list goes on and on...
```

Why naming is important?

Critical for Readability = Maintainability

Use the first few letters of the words


```
var fv string // flag value
```

Use fewer letters in smaller scopes

```
var bytesRead int // number of bytes read
```

```
var n int // number of bytes read
```

Use the complete words in larger scopes


```
package file
var fileClosed bool
```

Use mixedCaps like this


```
type PlayerScore struct
```

Use all capitals for acronyms

```
var localAPI string
```

```
var localApi string
```

Do not stutter


```
player.PlayerScore
```

❌

```
player.Score
```

✓

★★★

Do not use under_scores oR LIKE_THIS

★★★

```
const MAX_TIME int
```

❌

```
const MaxTime int
```

✓

```
const N int
```

✓